DAC/MAC/RBAC

- Owner-based discretionary access control (DAC)
  - Origins: academia
- Mandatory access control (MAC)
  - Origins: military
    - lattice-based access control (LBAC)
    - Bell-LaPadula (BLP) model
- Role based access control (RBAC)
  - Origins: business
CAUTION

- There is more to access control than
  - owner based DAC
  - MAC/LBAC/BLP
  - RBAC
- Generalized access control models
  - HRU, Take-Grant, SPM, TAM
  - Type enforcement
  - Generalized framework for access control

OWNER-BASED DAC

- owner has all-or-nothing power
  - superuser fallacy
- spaghetti of intent
- negative permissions make for messier spaghetti
- trojan horses can subvert intent
MAC/LBAC/BLP

- enforce one-directional information flow in a lattice of security labels
- can be used for
  - confidentiality
  - integrity
  - aggregation (Chinese Wall)
  - combinations of these

IMPASSE

<table>
<thead>
<tr>
<th>STRONG?</th>
<th>WEAK?</th>
</tr>
</thead>
<tbody>
<tr>
<td>WEAK?</td>
<td>STRONG?</td>
</tr>
</tbody>
</table>

Owner Based DAC

chaotic
anarchic
useless

MAC/LBAC/BLP

rigid
simple-minded
useless
RBAC

◆ A user’s permissions are determined by the user’s roles rather than
  ● user’s identity (DAC)
  ● user’s clearance (MAC)
◆ Facilitates
  ● administration of permissions
  ● articulation of policy

RBAC

◆ Policy neutral
◆ Policy oriented
  ● least privilege
  ● separation of duties
  ● encapsulation of primitive permissions
  ● separation of administration and access
  ● Roles are a semantic construct around which to build policy
RBAC vs GENERALIZED ACCESS CONTROL MODELS

- Generalized access control models can be configured to do RBAC
  - but
- do not provide convenient means for this purpose

RBAC is
- policy neutral
- policy oriented

General access control models are
- policy neutral
- mechanism oriented
RBAC

Two traditions

Support system-wide administrative functions

Programmed into individual applications

RBAC

- Often used to separate administrative functions
  - Operator
  - Auditor
  - Security Officer
  - User
RBAC: WHAT’S NEW

- Extend system support into application domain
- Use RBAC to manage RBAC

WHAT IS THE POLICY IN RBAC?

- RBAC is a framework to help in articulating policy
- The main point of RBAC is to facilitate security management
INTERACTION OF RBAC, MAC AND DAC

WHAT IS RBAC?

- multidimensional
- open ended
- ranges from simple to sophisticated
RBAC CONUNDRUM

- turn on all roles all the time
- turn on one role only at a time
- turn on a user-specified subset of roles

RBAC96 FAMILY

- RBAC0: VANILLA RBAC
- RBAC1: ROLE HIERARCHIES
- RBAC2: CONSTRAINTS
- RBAC3: ROLE HIERARCHIES + CONSTRAINTS
RBAC0

- **USER-ROLE ASSIGNMENT**
- **PERMISSION-ROLE ASSIGNMENT**

**USERS** ↔ **ROLES** ↔ **PERMISSIONS**

**SESSIONS**

**PERMISSIONS**

- **Primitive permissions**
  - read, write, append, execute
- **Abstract permissions**
  - credit, debit, inquiry
PERMISSIONS

- System permissions
  - auditorObject permissions
  - read, write, append, execute, credit, debit, inquiry

- Permissions are positive
- No negative permissions or denials
  - negative permissions and denials can be handled by constraints
- No duties or obligations
  - outside scope of access control
ROLES AS POLICY

◆ A role brings together
  ● a collection of users and
  ● a collection of permissions
◆ These collections will vary over time
  ● A role has significance and meaning
    beyond the particular users and
    permissions brought together at any
    moment

ROLES VERSUS GROUPS

◆ Groups are often defined as
  ● a collection of users
◆ A role is
  ● a collection of users and
  ● a collection of permissions
◆ Some authors define role as
  ● a collection of permissions
USERS

- Users are
  - human beings or
  - other active agents
- Each individual should be known as exactly one user

USER-ROLE ASSIGNMENT

- A user can be a member of many roles
- Each role can have many users as members
SESSIONS

- A user can invoke multiple sessions
- In each session a user can invoke any subset of roles that the user is a member of

PERMISSION-ROLE ASSIGNMENT

- A permission can be assigned to many roles
- Each role can have many permissions
MANAGEMENT OF RBAC

◆ Option 1:
  USER-ROLE-ASSIGNMENT and PERMISSION-ROLE ASSIGNMENT can be changed only by the chief security officer
◆ Option 2:
  Use RBAC to manage RBAC

RBAC1

ROLE HIERARCHIES

USER-ROLE ASSIGNMENT

PERMISSION-ROLE ASSIGNMENT

USERS

ROLES

PERMISSIONS

SESSIONS
HIERARCHICAL ROLES

Primary-Care Physician

Specialist Physician

Physician

Health-Care Provider

HIERARCHICAL ROLES

Supervising Engineer

Hardware Engineer

Software Engineer

Engineer
PRIVATE ROLES

Hardware Engineer’ Supervising Engineer Software Engineer’

Hardware Engineer Software Engineer

Engineer

RBAC: R3

ROLE HIERARCHIES

USER-ROLE ASSIGNMENT PERMISSIONS-ROLE ASSIGNMENT

USERS ROLES PERMISSIONS

SESSIONS CONSTRAINTS
CONSTRANTS

◆ Mutually Exclusive Roles
  ● Static Exclusion: The same individual can never hold both roles
  ● Dynamic Exclusion: The same individual can never hold both roles in the same context

CONSTRANTS

◆ Mutually Exclusive Permissions
  ● Static Exclusion: The same role should never be assigned both permissions
  ● Dynamic Exclusion: The same role can never hold both permissions in the same context
CONSTRANTS

◆ Cardinality Constraints on User-Role Assignment
  ● At most k users can belong to the role
  ● At least k users must belong to the role
  ● Exactly k users must belong to the role

◆ Cardinality Constraints on Permissions-Role Assignment
  ● At most k roles can get the permission
  ● At least k roles must get the permission
  ● Exactly k roles must get the permission
SCALE AND RATE OF CHANGE

- Hundreds of roles
- Thousands of users
- Frequent changes to
  - user-role assignment
  - permission-role assignment
- Less frequent changes for
  - role hierarchy